Model Transformation for a System of Systems Dependability Safety Case

by

J. Murphy & S. Driskell

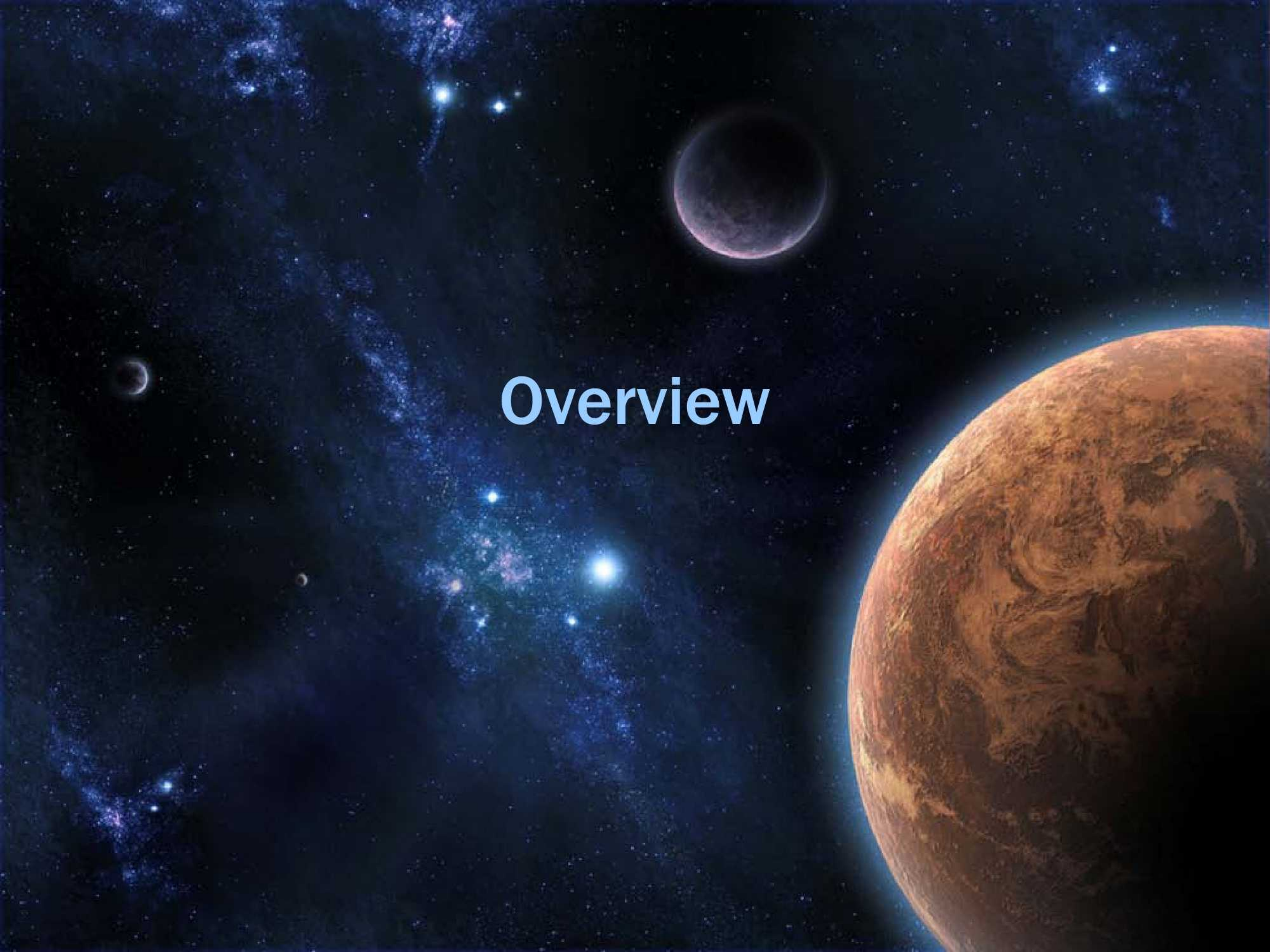
Presented by

Judy Murphy

jmurphy@mpl.com

Agenda

- **Overview**
  - – Dependability & Safety Overview
  - – Who, What, Why
  - – Engineering Safety Process
  - – Application to Non-Space Environment
- **Phase I Dependability & Safety**
  - – Where We Started
- **Phase II Dependability & Safety**
  - – Where we are today
  - – Model Transformation
- **Conclusion**



Overview

Acknowledgement and Disclaimer

- This research was sponsored by the NASA IV&V Facility
- The views and conclusions in this talk are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of the U.S. Government

http://www.nasa.gov/centers/ivv/home/

http://www.mpl. com/

http://www.tasc.com/

Dependability and Safety Effort

- **Address the need to identify safety-critical software requirements along with corresponding faults so that potential hazards may be mitigated early in the development of a System of Systems (SoS)**
- **Provides a proactive approach to the independent validation of safety requirements for systems of systems**
- **Provides a reusable set of artifacts for any family of spacecraft**
- **Provides a philosophy that can be applied to any industry**
- **Approach**
  - Move away from mission specific device fault conditions
  - Identify, compare and contrast subsystems
  - Create fault models based on functionality vs device functionality
- **Multi-phase project**
  - Phase I – Initial mission specific dependability and safety case
  - Phase II – Creation of generic fault conditions for cruise/orbit
  - Phase III – Creation of fault conditions for experiments and
  - Phase IV – Creation of fault conditions for surface operations for planetary robotic missions

Are all hazards identified and mitigated

Who, What, Why

- The mission of NASA’s IV&V program, under the auspices of the NASA Office of Safety and Mission Assurance (OSMA), is to provide the highest achievable levels of assurance for mission- and safety-critical software. The NASA IV&V Program provides assurance to our stakeholders and customers that NASA's mission-critical software will operate dependably and safely
- The NASA IV&V Program is building upon Phase 1 of spacecraft safety case study for a reusable set of artifacts for fault identification
- Mission success and spacecraft safety are both improved through contingency hazard management and the resulting failure risk reduction

Safety Engineering Process

- Starts with the system safety engineering activities to identify potential hazards and safety-critical functions, which are then traced through design into safety-critical hardware and software functions.
- Ends with validation and verification (V&V) of derived software safety requirements for controlling the hazard causal factors
- Team of software engineers, who are not the members of the development team, are tasked to validate and verify the SoS's software and requirements

Application to Non-Space Environment

- **Business models and strategies for product l**

Spacecraft Families

Communication

Science

Remote sensing, etc

THINK

Product Lines

Cell phones, Computers

Medical devices, Cars

Financial products, etc

- **E c valuation of pr s**

Spacecraft

Successful launch

Successful pay load deploy

Successful science collection

THINK

Product Lines

New iPhone® launch - sales

Windows Vista ® vs Windows 7 ® - sales

American vs foreign cars, etc

Application to Non-Space Environment

- **Organizational and process designs for product l**

Spacecraft Processes

NASA standards

MIL- STD -498

V-Model, CMMI, IEEE, etc

THINK

Product Line Processes

CMMI, Six Sigma, Agile

Regulatory agency rules/regulations

IEEE, etc

- **Service systems & their implications for product li**

Spacecraft Services

Fault management

Telemetry downlink

Command handling

Experiment control

THINK

Product Line Services

Cell phone alerts & applications

Interface design

Customer data access

Online selling

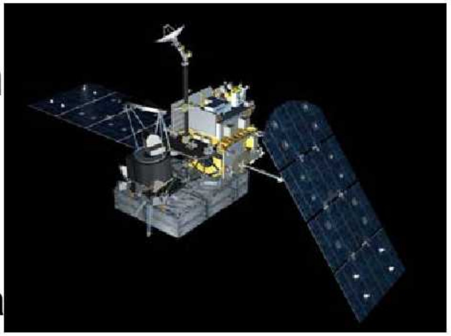
Onstar ®

We are not that different

Phase I Overview

Where We Started

- **Built a dependability and safety case for safe-hold**
  - Global Precipitation Measurement (GPM) mission
    - Studies global precipitation
  - Autonomous software for managing spacecraft hazards without ground intervention
    - Are all subsystem faults requiring safe-hold included in sa hold monitor?
    - Are all safe-hold requirements identified?

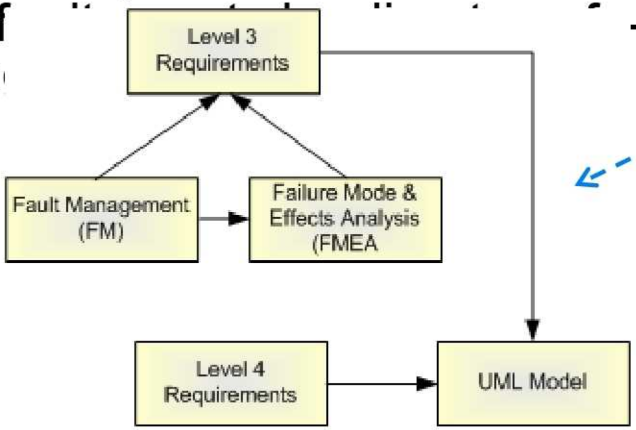


- **The IV&V analyses are model-based, striving to obtain goodness of product data in terms of three questions:**
  - *What is the system software supposed to do?*
  - *What the system software is not supposed to do?*
  - *What is the system software's expected response under adverse conditions?*

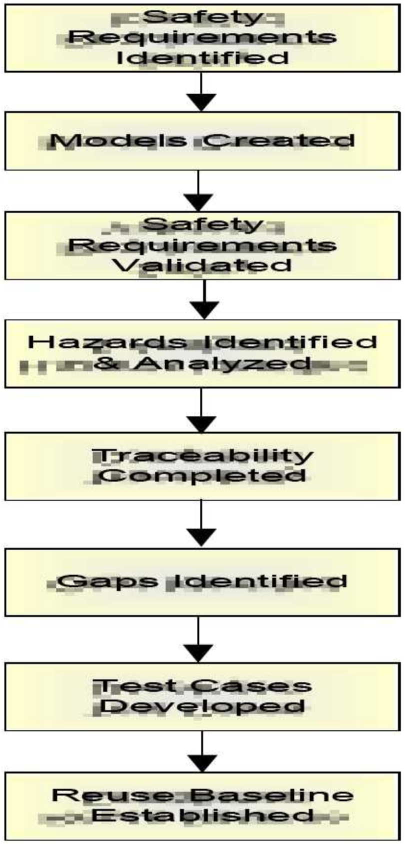
Build a dependability & safety case for SoS testing

Process Creation

- **Created a new IV&V analysis process**
  - Started with an IV&V developed *independent* list of fault conditions specific to GPM
  - Based on previous mission experience and GPM knowledge
  - Used to help determine if there were gaps in the f -hold and missin



High-level dependability & safety case [2]

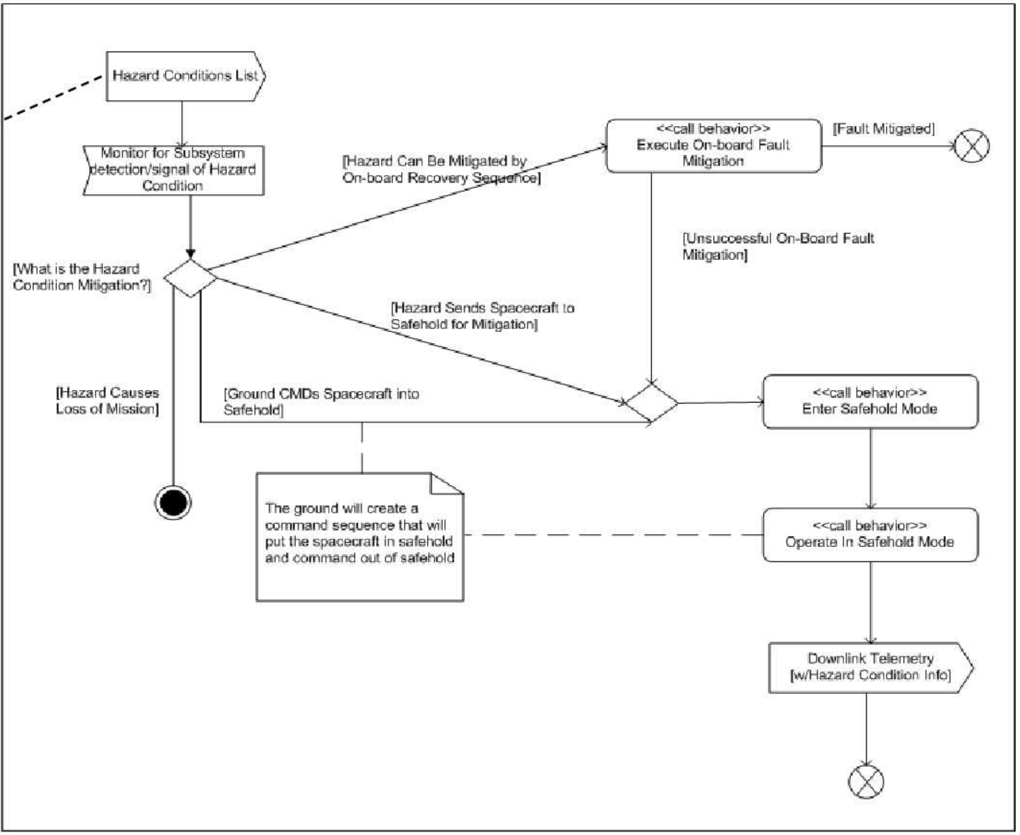

Safety Requirements Identified
Models Created
Safety Requirements Validated
Hazards Identified & Analyzed
Traceability Completed
Gaps Identified
Test Cases Developed
Reuse Baseline Established

IV&V analysis process [2]

The right process identifies missing requirements
Process and artifacts are reusable

Sample SRM Artifact

- **Mission specific safe-hold activity diagram for fault management**


Hazard Conditions List
Monitor for Subsystem detection/signal of Hazard Condition
[What is the Hazard Condition Mitigation?]
[Hazard Can Be Mitigated by On-board Recovery Sequence]
<<call behavior>> Execute On-board Fault Mitigation
[Fault Mitigated]
[Unsuccessful On-Board Fault Mitigation]
[Hazard Sends Spacecraft to Safehold for Mitigation]
[Hazard Causes Loss of Mission]
[Ground CMDs Spacecraft into Safehold]
<<call behavior>> Enter Safehold Mode
The ground will create a command sequence that will put the spacecraft in safehold and command out of safehold
<<call behavior>> Operate In Safehold Mode
Downlink Telemetry [w/Hazard Condition Info]

SUBSYSTEMS - partial list

Command & Data Handling
- Power supply part/connector failure
- RAD 750 part/connector failure
- Bulk memory part/connector failure
- Temperature & analog part/connector failure
- Payload a GPS part/connector failure

Safety Mech. & Attitude Control
- Propulsion I/F part/connector failure
- Solar array & high gain antenna I/F part/ connector failure
- Attitude sensors and actuators I/F part/ connector failure

Guidance, Navigation & Control
- Star tracker part/connector failure
- Sun sensor part/connector failure
- Inertial reference unit part/connector failure
- Magnetometer part/connector failure
- Reaction wheel part/connector failure
- Global positioning system part/connector failure

Electrical Power Systems
- Power monitor & control part/connector failure
- Battery part/connector failure
- Survival heater part/connector failure
- Subsystem I/F part/connector failure
- Instrument I/F part/connector failure

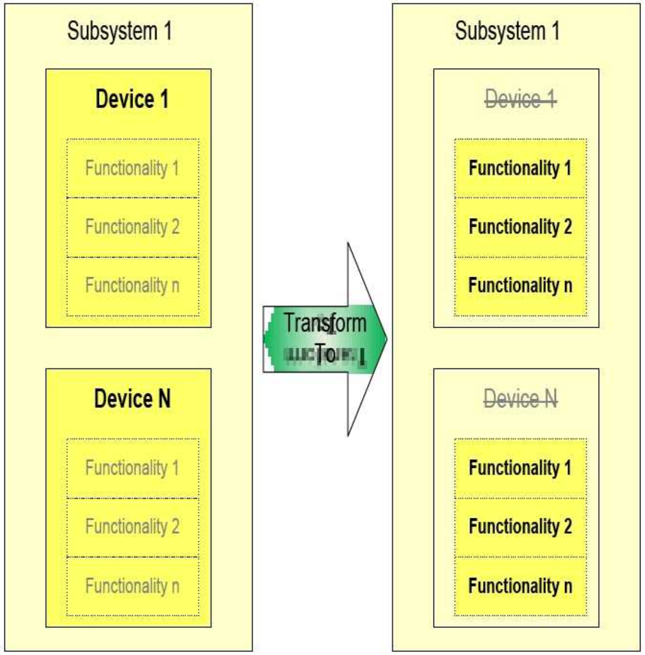
Phase I End Result

- Ensure these hazards are managed and failure risk is reduced
- Deliver a reusable standardized spacecraft software safety case for IV&V
- Identify missing safe-hold requirements
- Provide software test scenarios
- IV&V efforts on other science missions have decided to build safety cases using this process
- This approach will be applied to other behaviors besides safe-hold
- Mapped IV&V first science list of fault conditions to Mars Science Laboratory (MSL) ; Fault and Failure Analysis (FFA) data
  - MSL FFA data is at a different level than the IV&V list of fault conditions, so activity was partially successful
  - This is being addressed in study Phase II

Phase II Overview

Model Transformation From Specific to Generic

- **Moving from specific faults to generic faults**
  - Faults are currently ***device dependent*** not functionality dependent
  - Faults are not always obviously or easily reusable on other missions
- **Families of spacecraft may use the same underlying architecture**
  - Subsystem device names are often different
- **Create models and fault conditions based on the *functionality* of a subsystem at the highest level**
- **Created a process to go from specific generic**

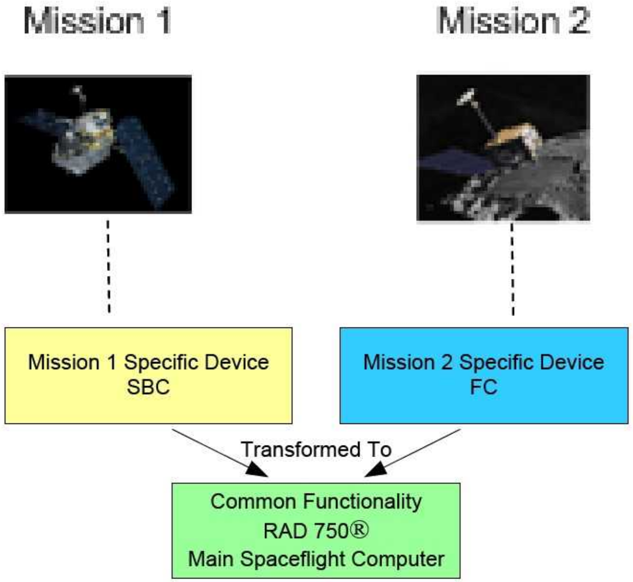


Focus on functionality – not devices

Model Transformation

Identify Common Functionality

- **Compare space missions to each other**
  - Share many of the same characteristics
  - All space missions have subsystems that deal with
    - Telemetry, command and data handling, guidance navigation and control, 1553 bus, temperatures, voltages, etc
  - Functionality of other missions uses pyrotechnics, robotic rovers and unique experiments
  - Those subsystems may have differing designs and device names, but the subsystem functionality is the common thread



Mission 1

Mission 2

Mission 1 Specific Device
SBC

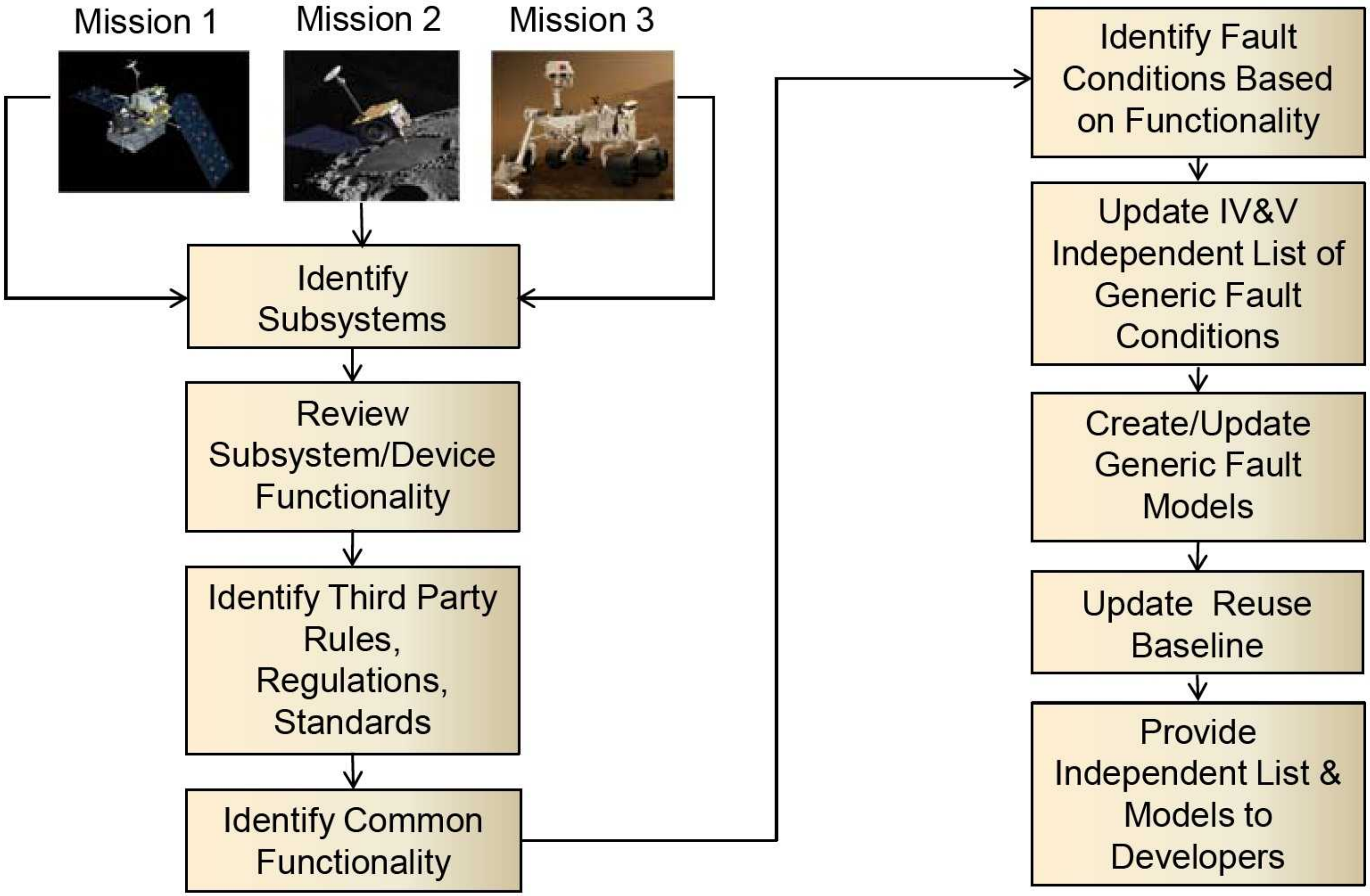
Mission 2 Specific Device
FC

Transformed To

Common Functionality
RAD 750®
Main Spaceflight Computer

Important - find common functionality

Model Transformation Process



Mission 1

Mission 2

Mission 3

Identify Subsystems

Review Subsystem/Device Functionality

Identify Third Party Rules, Regulations, Standards

Identify Common Functionality

Identify Fault Conditions Based on Functionality

Update IV&V Independent List of Generic Fault Conditions

Create/Update Generic Fault Models

Update Reuse Baseline

Provide Independent List & Models to Developers

Reusable process

Specific Safe-hold Fault Management Example

SUBSYSTEMS - partial list

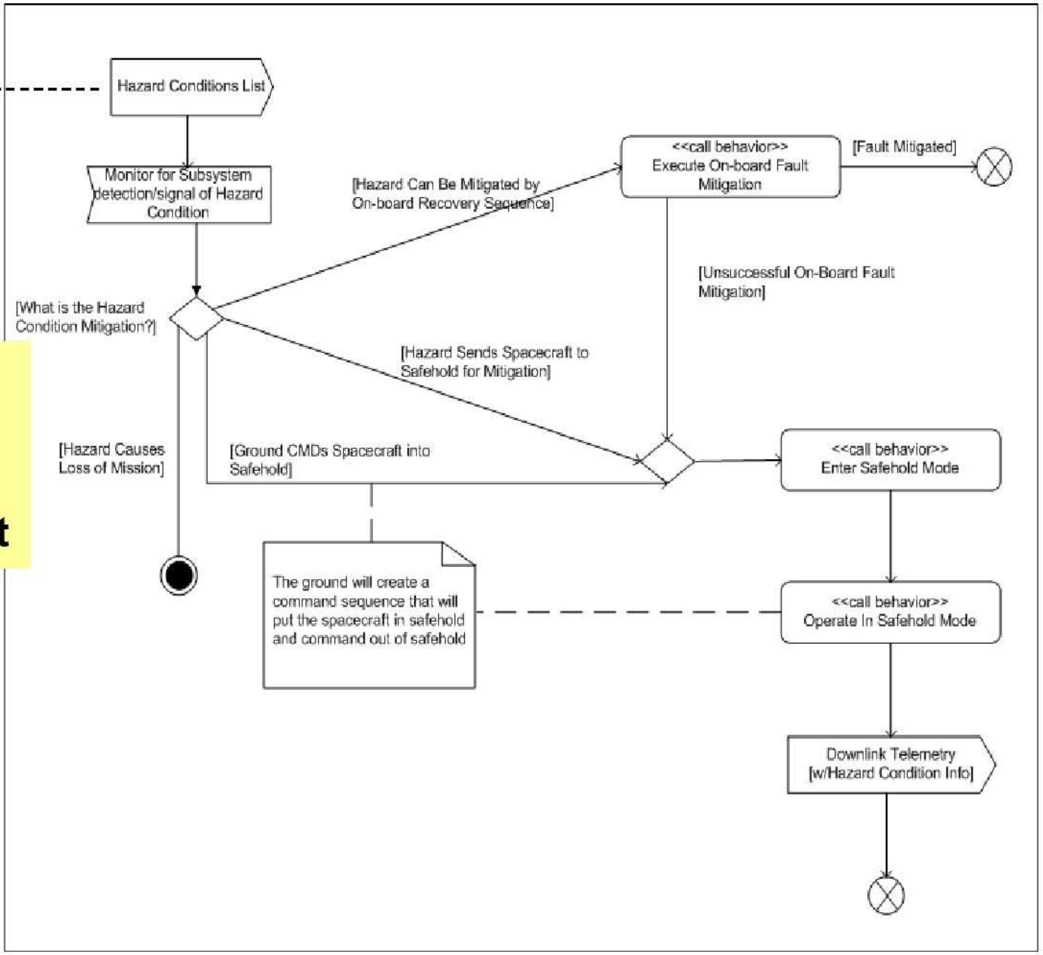
Command & Data Handling
- Power supply part/connector failure
- RAD 750 part/connector failure
- Bulk memory part/connector failure
- Temperature & analog part/connector failure
- Payload a GPS part/connector failure

Safety Mech. & Attitude Control
- Propulsion I/F part/connector failure
- Solar array & high gain antenna I/F part/connector failure
- Attitude sensors and actuators I/F part/connector failure

Guidance, Navigation & Control
- Star tracker part/connector failure
- Sun sensor part/connector failure
- Inertial reference unit part/connector failure
- Magnetometer part/connector failure
- Reaction wheel part/connector failure
- Global positioning system part/connector failure

Electrical Power Systems
- Power monitor & control part/connector failure
- Battery part/connector failure
- Survival heater part/connector failure
- Subsystem I/F part/connector failure
- Instrument I/F part/connector failure

Mission device name dependent


Hazard Conditions List
Monitor for Subsystem detection/signal of Hazard Condition
[What is the Hazard Condition Mitigation?]
[Hazard Can Be Mitigated by On-board Recovery Sequence]
<<call behavior>> Execute On-board Fault Mitigation
[Fault Mitigated]
[Unsuccessful On-Board Fault Mitigation]
[Hazard Sends Spacecraft to Safehold for Mitigation]
[Hazard Causes Loss of Mission]
[Ground CMDs Spacecraft into Safehold]
<<call behavior>> Enter Safehold Mode
The ground will create a command sequence that will put the spacecraft in safehold and command out of safehold
<<call behavior>> Operate In Safehold Mode
Downlink Telemetry [w/Hazard Condition Info]

Less flexible, less reusable

Generic Fault Management Example

SUBSYSTEMS - Capability/functionality issues - partial list

Command & Data Handling
- Main spaceflight computer HW/SW issue
- Temperature & analog HW/SW issue
- Payload & GPS Subsystem I/F HW/SW issue
- 1553 I/F HW/SW issue
- Serial bus I/F HW/SW issue

Safety Mech. & Attitude Control
- Propulsion I/F HW/SW issue
- Solar array & antenna I/F HW/SW issue
- Attitude sensors and actuators I/F HW/SW issue
- 1553 I/F HW/SW issue
- Serial bus I/F HW/SW issue

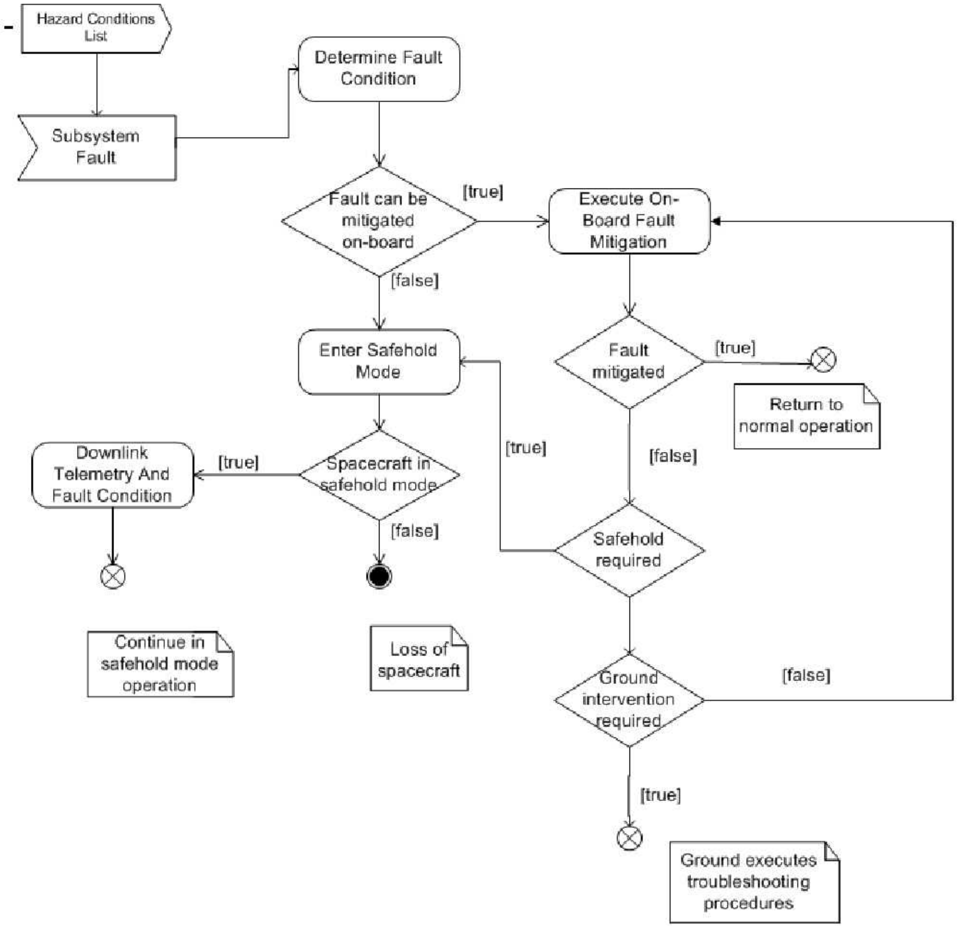
Guidance, Navigation & Control
- Star tracking HW/SW issue
- Sun sensing HW/SW issue
- Inertial references HW/SW issue
- Magnetometer HW/SW issue
- Reaction wheel HW/SW issue
- Global positioning HW/SW issue
- 1553 I/F HW/SW issue
- Serial bus I/F HW/SW issue

Electrical Power
- Power monitoring & control HW/SW issue
- Battery HW/SW issue
- Survival heater HW/SW issue
- Subsystem 1 I/F part/connector failure
- Subsystem N I/F part/connector failure
- Instrument/experiment I/F HW/SW issue
- 1553 I/F HW/SW issue
- Serial bus I/F HW/SW issue

Subsystem N
- Functionality 1 HW/SW issue
- Functionality N HW/SW issue

Functionality Dependent

Device independent



Hazard Conditions List

Subsystem Fault

Determine Fault Condition

Fault can be mitigated on-board

[true]

Execute On-Board Fault Mitigation

[false]

Enter Safehold Mode

Fault mitigated

[true]

Return to normal operation

[false]

Spacecraft in safehold mode

[true]

Downlink Telemetry And Fault Condition

Continue in safehold mode operation

[false]

Loss of spacecraft

[true]

Safehold required

[false]

Ground intervention required

[false]

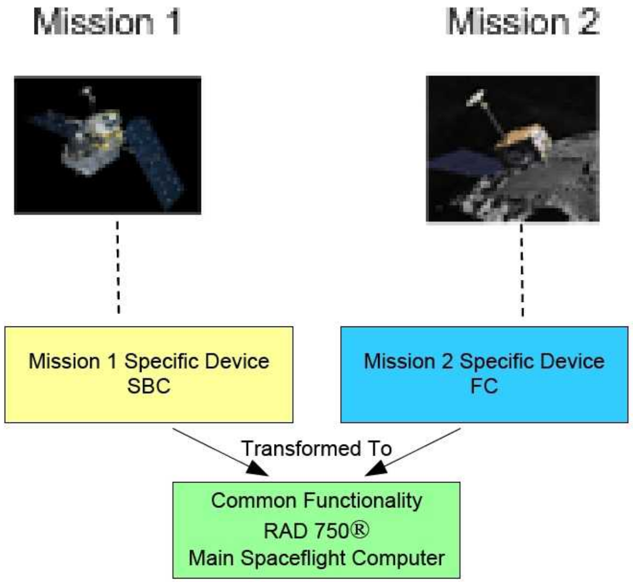
[true]

Ground executes troubleshooting procedures

Transformation - reusable on any mission

Generic Main Spaceflight Computer Example

- **Understand subsystem functionality**
  - Decompose into known and potential hardware and software faults
    - Peripheral Component Interconnect (PCI) status register errors
    - Excessive accumulation of uncorrectable SDRAM memory errors
    - Overcurrent/undercurrent
    - Overvoltage/undervoltage
    - CPU halt/hung
    - Etc


Mission 1
Mission 2
Mission 1 Specific Device
SBC
Mission 2 Specific Device
FC
Transformed To
Common Functionality
RAD 750®
Main Spaceflight Computer

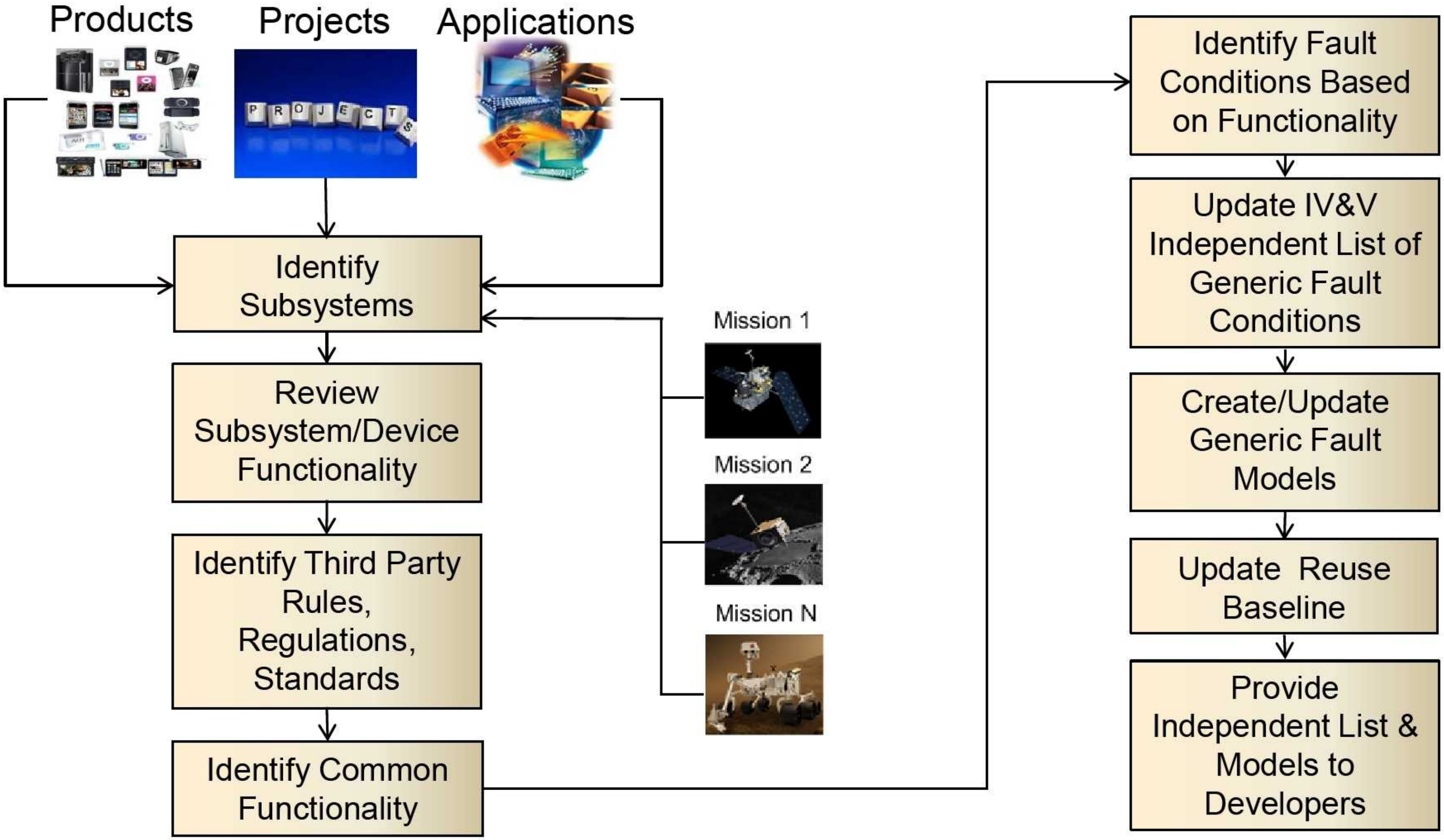
Compare, contrast, analyze

Applying Phase II To Your Project

- **Transformation from the specific to the generic**
  - – Think product lines – not spacecraft
  - – Apply the model transformation process
  - – Replace the space mission examples with your system information
  - – Decompose the system into subsystems
    - Look at projects, programs, applications, services, etc
    - Focus on functionality
  - – Create models
  - – Add lessons learned from previous projects
  - – Establish a baseline

Reusable in any environment

Applying Phase II To Your Project



Products

Projects

Applications

Identify Subsystems

Review Subsystem/Device Functionality

Identify Third Party Rules, Regulations, Standards

Identify Common Functionality

Mission 1

Mission 2

Mission N

Identify Fault Conditions Based on Functionality

Update IV&V Independent List of Generic Fault Conditions

Create/Update Generic Fault Models

Update Reuse Baseline

Provide Independent List & Models to Developers

Reusable process

Conclusion

- **Identified a proactive approach using reusable fault conditions - based on functionality**
- **Phase II introduces a new way to independently validate software safety requirements, via the comparison of the fault management artifacts against the IV&V team's own list of fault conditions – based on functionality**
- **Helps the mission developer ensure they have identified the correct fault conditions & identifies missing requirements**
  - Promotes feedback from the developer
- **Builds a foundation for dependability and safety that is reusable**
  - Applicable in any environment

References

[1] US Government Accountability Office, NASA: Assessments of Selected Large-Scale Projects, Report Number GAO-10-227SP, February 1, 2010. Accessed on 29, March 2010: http://www.gao.gov/products/GAO-10-227SP.

[2] S. Driskell, J. Murphy, J.B. Michael and M. Shing. “Independent Validation of Software Safety Requirements for Systems of Systems,” *Proc. 2010 IEEE International Conference on System of Systems Engineering*, Loughborough University, UK, 22-24 June 2010.

[3] NASA Software Safety Standard (w/Change 1 dated 7/08/04), NASA-STD-8719.13B, March 12, 2008. Accessed on 14, November 2010: http://www.hq.nasa.gov/office/codeq/doctree/871913.htm.

[4] NASA General Safety Program Requirements (w/Change dated 7/20/09), NPR 8715.3C, March 12, 2008. Accessed on 14, November 2010: http://nodis3.gsfc.nasa.gov/npg_img/N_PR_8715_003C_/N_PR_8715_003C_.pdf.

[5] W. Gibbs, “Software’s Chronic Crisis,” *Scientific American*, pp. 86, Sep. 1994.

[6] NASA Software Safety Guidebook (w/Change 1 dated 7/08/04), NASA-GB-8719.13, March 31, 2004. Accessed on 14, November 2010: http://www.hq.nasa.gov/office/codeq/doctree/871913.htm

[7] PCI Local Bus specification, Revision 2.2, December 18, 1998. Accessed on 19, November 2010: http://www.ece.mtu.edu/faculty/btdavis/courses/mtu_ee3173_f04/papers/PCI_2d2.pdf

[8] “NASA IV&V Vision and Mission.” NASA IV&V Facility, accessed 9, December 2010, http://www.nasa.gov/centers/ivv/about/visionmission.html.